PENNY – PERSONAL SECURITY ASSISTANT

A Thesis
Presented to
The Academic Faculty

by

Ilya Golod

In Partial Fulfillment
of the Requirements for the Degree in
Computer Science in the
College of Computing, Georgia Institute of Technology

Georgia Institute of Technology
December 2019

TABLE OF CONTENTS

LIST OF FIGURES AND TABLES

ABSTRACT

This study presents Penny, a virtual assistant that monitors various parameters and conditions of the user's machine and notifies them in case it senses a potential vulnerability. Penny also provides the user with information on the possible ways of eliminating or avoiding the vulnerability. The purpose of the study is to find out how specific types of security-related communication with end-users by a virtual assistant like Penny can positively affect users' security-related behaviors.

SECTION 1

INTRODUCTION

Despite the significant advances in cybersecurity [3, 8], many people still fall victim to numerous attacks. This can be credited to their inability or reluctance to use the modern cybersecurity capabilities to protect their machines. The significance of the issue manifests itself in various situations. Due to the leaks of credit card numbers, many users to lose their money every day. Lots of private files get stolen just because of disabled two-factor authentication. One of the main goals of studies in usable security and privacy is to prevent such situations by making the cybersecurity features more understandable or easy to use, thus increasing the likelihood of users engaging in safe security behaviors [3, 5, 8].

There are many angles from which one can approach this problem. Some researchers determined the influence of social factors on users' security-related decision making by exposing them to the trends in the security decisions that other people make and analyzing how it changes such decisions of the users [2, 3]. Redmiles *et al.* and Herley *et al.* designed mathematical models explaining why the users do or do not make security decisions in certain situations [4, 5]. At the same time, separately and so far quite unrelated to the field of usable security and privacy, there has been quite a large amount of research and work done to this day in development of personal virtual assistants [1], which are becoming more and more popular nowadays.

The goal of our team is to examine the potential synergy of these two areas. We are developing a virtual assistant, Penny, whose job is to monitor the condition of the user's machine and to proactively notify them in case she finds anything that can be

considered a potential vulnerability. The information Penny provides to the end-user will include the description of the vulnerability and potential threats it may present as well as the possible ways of resolution. Considering it has been shown that awareness and understanding of security issues has a positive impact on making rational security decisions [6], we hope that such an assistant can increase the chances of a user engaging in safe personal security behaviors. Furthermore, since social factors (i.e. exposure to the security-related choices of others) have already been demonstrated to influence users' security decision making [2, 3], we want to find out if pseudo-social interactions with a virtual assistant can help to achieve a similar effect. The ultimate goal of our research is to develop Penny and conduct a user study to analyze the effect of exposure to a virtual assistant like her on the security-related behavior of end-users.

SECTION 2

LITERATURE REVIEW

Considering the rising levels of computerization of various aspects of lives of modern people, the problem of usability of security tools and features is relevant nowadays as the significance of people's cybersecurity is becoming comparable with the obvious importance of their physical security. Issues associated with usability of personal cybersecurity have been approached by different researchers from a variety of angles, some of which are discussed in this section.

One of the particularly interesting approaches revolves around an attempt to tie together users' security and social factors. A successful study by Das *et al.* determining the importance of social aspect in increasing users' awareness of security issues was presented in their paper in 2014 [2]. In this study, randomly selected Facebook users were shown different security announcements. Some of these announcements had information on how many friends of a user were already using a certain security feature, and some had no socially oriented section in them. The work demonstrated that the announcements employing social proof drove 37% more users to explore security features, thus showing that just displaying the number of friends using the security features to a user can help significantly in raising the awareness of such features. In another paper [3], via interviews and quantitative data analysis, Das *et al.* found a strong connection between social processes and users' awareness of, motivation to use and knowledge of how to use security and privacy tools and features. They also distinguished two main goals of a security-related conversation: to warn or protect others from an immediate threat or to

gather information about solving a privacy problem. Observability of a security feature was defined by them as the key enabler of socially triggered security behavior change.

Quantification and rationalization of users' behavior constitute another significant portion of research conducted in the field of usable security. For instance, in their work [4], Redmiles *et al.* tried to model a situation where a user is forced to make a potentially meaningful security decision to measure the cost and utility of adopting a security behavior given the time spent on the execution of the latter and some external data such as the participant's wage. An experiment was set up where the user has to create a bank account on a website, and they can either enable two-factor-authentication for it or not. The user is notified of the potential risk of losing their money beforehand. More than 50% of the participants of the experiment were found to make utility optimal—in terms of cost and benefit—decisions in this setting. Redmiles *et al.* have also found that users' decisions can be modeled quite well as a function of their past behavior and other parameters such as the knowledge of costs and awareness of risks and context. This work strengthened the claim previously made by Herley that users tend to make much more rational security decisions than it may seem by demonstrating a clear tendency towards optimization in such decisions [5]. These findings imply that focus of the work of usable security engineers and researchers should be shifted from making the users make rational decisions to making these decisions actually rational. In our research, we want to see how an interaction with a virtual assistant, which will explain the rationale behind the specific security-related decisions to the end-user, will help users better understand which actions are in fact rational.

Users' awareness of the outcomes and the supposed rationale of their security decisions is also treated as one of the key factors in improving the usability of security. We cannot guarantee that a professional researcher or engineer will always be able to put themselves into the place of the end-user and try to look at the situation from their perspective and level of awareness. This problem is discussed in detail in the work by Abu-Salma *et al.*, which presents a study of end-users' perception of secure communication tools and their security properties, and how certain mental models (i.e. erroneously applying telephony-related concepts to digital communication) underpin people's beliefs [8]. Through an analysis of the data collected from interviews, they proved that a significant part of users does not have a comprehension of the essential concepts of protection the secure communication tools offer. One of the main arguments proposed by the study is that there is a strong necessity to understand how majority of the users perceive security of communication and what makes them decide to whether or not to adopt secure tools, in order to design and build communication tools that protect user. Many other researchers also distinguish increasing users' awareness of specific consequences of their decisions as one of the most important aspects of user interaction to take into consideration for usable security tools implementation [6, 7]. Our research, in turn, is focused on examining a specific application of these ideas through a virtual assistant.

Independently of the listed works on usable security, substantial progress has been made in the research on implementation of personal virtual assistants. One of the most prominent examples here is Almond, an open, secure, and programmable virtual assistant for various online services (i.e. social networks, mailboxes, etc.) and IoT-devices,

described by Campagna *et al.* [1]. The work addresses four major challenges in virtual assistant technology: generality, interoperability, privacy and usability – and provides an example of approaches to solving each of them in a form of a prototype of an actual multi-functional interactive assistant.

In our study, we want to try approaching the problem of enhancing the effectiveness of users' interaction with security tools from a slightly different angle. We want to integrate the data from the previous research on factors impacting users' security decisions into a virtual assistant. Our team is designing an assistant, which will proactively notify users about potential vulnerabilities of their machines and provide them with necessary information about these vulnerabilities. In this way, we want to combine pseudo-social interaction with giving the users an opportunity to make an informed decision in a given situation. As there has been no substantial research conducted yet on potential influence of virtual assistant on end-users' security-related decision making, we are trying to examine this question in detail. Specifically, our goal is to analyze whether such an assistant installed on a user's device will enhance the quality of their security-related decisions.

SECTION 3

METHODS AND MATERIALS

Before we can conduct the analysis that we are aiming for, we need to implement our assistant itself. For the technical implementation of Penny, we attempted to use the technology stack that would allow maximum potential flexibility in case we want to change some parts of it in the future for the needs of the research. For this purpose, we chose our system to have a modular structure [Fig. 1]. The core of Penny—her business logic and vulnerability detection framework—is contained within a local web server written in Python, so that it would be totally independent from the front-end and other components of the assistant. Then, there is a Chrome browser extension which internally operates independently of the server and communicates with the latter through local HTTP requests (we are targeting only one specific browser for research purposes to make the development process simpler). The extension is used for detection of vulnerabilities associated with Internet-related activities of the user. The next component of the structure of our assistant is its user interface. The one we have currently is implemented in JavaFX; however, we may add other implementations of the UI in the future to potentially make the comparison between effectiveness of them one of the research questions that we answer. Just like the extension, the user-facing part is integrated with the server through HTTP protocol and its implementation is almost fully independent of implementations of the other components of Penny.

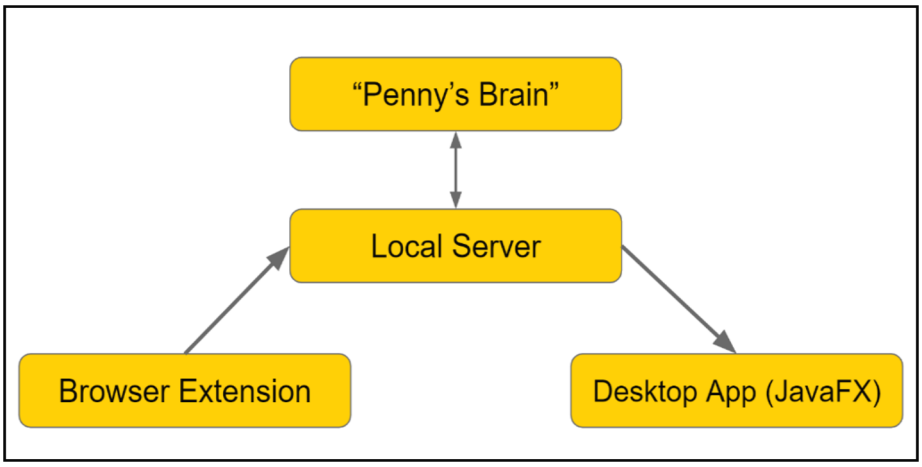

Figure 1. Architecture of Penny

To collect data on the effect of Penny’s interactions on users’ behavior, we obviously need her to be able to handle some actual security-related scenarios that can be encountered by the users in real life. Currently, Penny’s capabilities include but are not limited to detection of out-of-date software lacking patches against newly discovered vulnerabilities, detection of connections to insecure wireless networks, analysis of the state and presence of an antivirus on the user’s machine, identification of requests of sensitive information by third-party websites, etc.

The most developed—and, at the moment, the most promising—of the scenarios based on the aforementioned features is transmission of sensitive information through unencrypted wireless networks. An example of this scenario is a situation that can happen to virtually any modern-day user of a computer. A user is sitting in a, say, coffee shop and connects to its unencrypted Wi-Fi with their laptop. Then, they decide to go online to shop for some shoes. After selecting the nicest pair of shoes that they can find, they go to the checkout page of the online store to place the order and make a payment. To do the latter, they obviously need to enter their credit card information. As soon as they press the submit button, their credentials will be sent over the unencrypted Wi-Fi network and

will be readable by any local perpetrator who is patient and skillful enough to intercept the data. To prevent this, Penny constantly monitors the presence of sensitive information (such as credit card credentials, SSN number, etc.) entry boxes on the websites visited by the user. Whenever she detects such a box on a webpage, she checks if the user's machine is currently connected to an insecure wireless network. If so, she immediately notifies the user [Fig. 2] and, if asked, explains them the potential solutions to this issue (i.e. wait for an encrypted Wi-Fi to make the purchase).

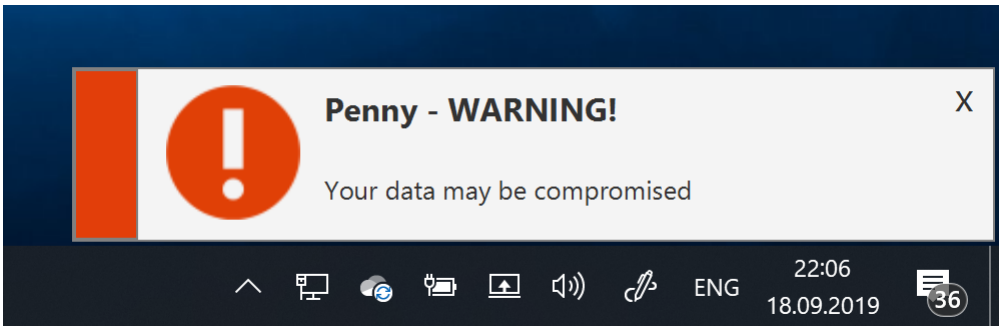

Figure 2. An example of Penny's warning notification

Furthermore, as one of its key features, Penny can elaborate more on the details of the vulnerability to give the user a better understanding of the rationale that stands behind Penny's recommendations.

The choice of this sensitive information scenario was not random. To determine the most efficient and relevant direction of the development we conducted a survey among college students asking them to pick their top-3 cybersecurity-related scenarios that they are most afraid of happening to them. Leaks of credit card information and SSNs turned out to be the top responses in this survey [Fig. 3]. It is, however, necessary to emphasize that this survey is not an official result of this study and was only used to generally determine the most relevant scenario to explore.

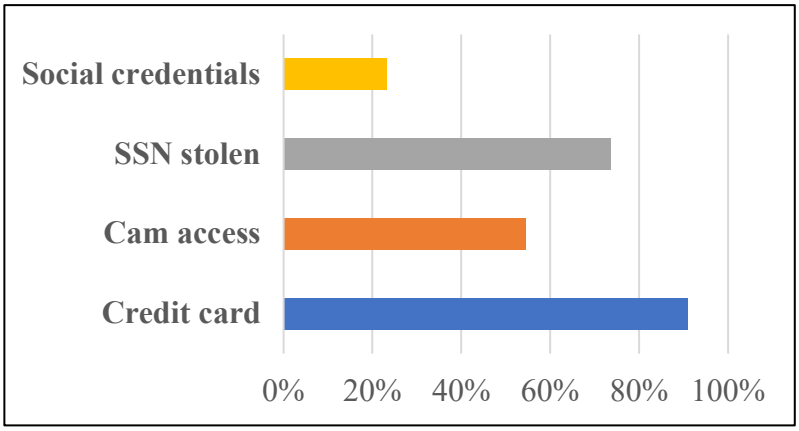

Figure 3. Four scenarios that users are most afraid of according to the survey

As for Penny's user interface, we wanted it to have a somewhat two-level structure. What is meant by this here is that users are able to interact with Penny quickly through a GUI (graphic user interface) based interfaces (the way we do with most of existing software) as well as through a chat-based interface where they can get answers on the questions they have about their security issues and some more detailed feedback in general. For example, if we come back to the credit card credentials leak scenario, as soon as the user gets Penny's notification regarding their problem, they have several options of the consequent interaction. First, they can just ignore or swipe away the notification (this is the kind of behavior we are attempting to minimize). Second, they can open Penny, read her concise description of the issue, understand everything, and either act accordingly to prevent the vulnerability or press the ignore button to silence Penny on this matter. Third, if the user does not understand why the issue should be addressed or how exactly they should act to prevent it, they can use the chat interface to aske Penny any questions they have [Fig. 4].

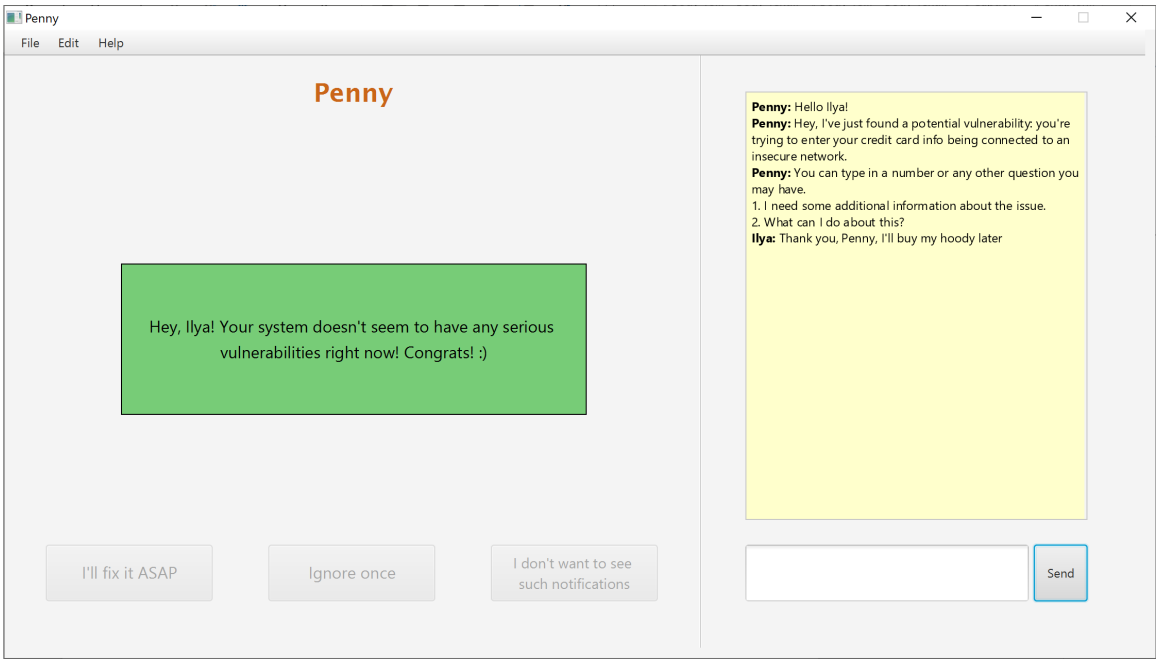

Figure 4. Mockup of Penny's two-layer interface

It would be of course quite beneficial to later change the aesthetics of Penny's interface in order to make the application more appealing to the user, but, since these kinds of cosmetic modifications are not exactly the scope of the research that we are conducting right now, we leave them as the future work. Aesthetic aspects would be quite important for public use of the assistant, but for a lab-based user study, we consider it less crucial.

The ultimate goal of our research is nevertheless not Penny's implementation itself but the user study that we want to conduct with it. Currently, we are aiming at conducting a lab-based study. We will invite recruited participants into the lab, let them sit in front of a computer and use Penny in several predesigned scenarios for a certain period of time. In the meantime, we will be, electronically and manually, collecting data and statistics on various aspects of users' interactions with our assistant. Despite this setup being currently our main one, it would bolster up the conclusions that we draw from our findings a lot if we conducted a field study where users are able to interact with

Penny in a natural setting. It would make the ideas delivered in the research somewhat more founded in terms of both their applicability to real-world situations. However, such a user study setting would be possible only in case a number of significant technical enhancements are made in Penny's implementation.

SECTION 4

RESULTS AND DISCUSSION

As I have said before, we are still working towards the user study after which we will have the data to analyze and will be able to draw corresponding conclusions from it. However, as we are moving closer to the final point of the research, we are obviously envisioning some of our work's possible applications as well as several issues that may arise depending on the specific content of our ultimate results.

In case we can see a significant improvement of users' security-related behavior under their exposure to Penny, it would mean that our approach may be a viable way of increasing the level of personal security in many different contexts. For example, incorporating such a social aspect—a virtual assistant specifically—into the operation of personal cybersecurity software (i.e. antivirus software) may result in an increase of its effectiveness. Results of our research, if positive, may in turn make this potential advantage of usage of virtual assistants and pseudo-social interactions more apparent to the producers of antivirus software, thus increasing the chances of such ideas to be implemented and raising the global level of personal cybersecurity.

Furthermore, the framework that we will have built for the detection of certain types of unsafe user behavior (i.e. unprotected interaction with online payment systems, etc.) can be applied to further research projects in our field of study both in our laboratory and beyond. We have been keeping such a potential application in mind while building it, making it as easy to use for outsiders as we can.

We have not yet decided on some of the specifics of our user study, but as we are inclining towards a lab-based study, we realize that even in case of a strong correlation

between our subjects' safe security-related behavior and their exposure to our virtual assistant, a question may arise regarding the applicability of these results to situations outside of a lab setting. In case we face such a problem, a significant part of the future work on this project will be focused on making it possible to conduct a field study which will give us a better representation of real-life performance of virtual assistants. Currently, it seems that there are two main aspects that we need to work on to achieve applicability of Penny to a field study. First, we need to improve Penny's visual design and general user experience to make the assistant just more pleasant and convenient to interact with in real life. Second, we need to integrate a statistics-gathering module into Penny's backend as this is crucial for collecting the results of any field study. Anyway, currently, we are aiming at a lab study as our closest milestone. From there, it will be easier to move forward to a field study if such a need arises.

However, we cannot of course guarantee at the current point, that the results of our user study will correlate well with what we expect them to be. There is a possibility that our subjects will show little or no improvement in their cybersecurity-related behaviors with presence of our assistant. In such a case, we will have to look for any issues in our premises or design choices. The problem might occur in the details of Penny's interaction with the user (i.e. format or wording of messages, types of notifications, etc.) as well as in our choice of specific security scenarios to address in the study. Finally, it may just be the case that, in opposition to our expectations, a virtual assistant is not a feasible way of making a positive impact on personal users' security behavior.

In any way, as we have neither positive nor negative results yet, we are doing our best to minimize the chance of our design choices to negatively affect the conclusiveness of our user study by making them as thoroughly adjusted for the study as we can. Our future steps, in turn, as described above, will depend entirely on the content of the study's result.

REFERENCES

1. Campagna, Giovanni, Rakesh Ramesh, Silei Xu, Michael Fischer, and Monica S. Lam. "Almond: The architecture of an open, crowdsourced, privacy-preserving, programmable virtual assistant." In *Proceedings of the 26th International Conference on World Wide Web*, pp. 341-350. International World Wide Web Conferences Steering Committee, 2017.

2. Das, Sauvik, Adam DI Kramer, Laura A. Dabbish, and Jason I. Hong. "Increasing security sensitivity with social proof: A large-scale experimental confirmation." In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, pp. 739-749. ACM, 2014.

3. Das, Sauvik, Tiffany Hyun-Jin Kim, Laura A. Dabbish, and Jason I. Hong. "The effect of social influence on security sensitivity." In *10th Symposium On Usable Privacy and Security ({SOUPS} 2014)*, pp. 143-157. 2014.

4. Redmiles, Elissa M., Michelle L. Mazurek, and John P. Dickerson. "Dancing Pigs or Externalities?: Measuring the Rationality of Security Decisions." In *Proceedings of the 2018 ACM Conference on Economics and Computation*, pp. 215-232. ACM, 2018.

5. Herley, Cormac. "So long, and no thanks for the externalities: the rational rejection of security advice by users." In *Proceedings of the 2009 workshop on New security paradigms workshop*, pp. 133-144. ACM, 2009.

6. Forget, Alain, Sarah Pearman, Jeremy Thomas, Alessandro Acquisti, Nicolas Christin, Lorrie Faith Cranor, Serge Egelman, Marian Harbach, and Rahul Telang. "Do or do not, there is no try: user engagement may not improve security outcomes." In *Twelfth Symposium on Usable Privacy and Security ({SOUPS} 2016)*, pp. 97-111. 2016.

7. Mathur, Arunesh, Josefine Engel, Sonam Sobti, Victoria Chang, and Marshini Chetty. "" They Keep Coming Back Like Zombies": Improving Software Updating Interfaces." In *SOUPS*, pp. 43-58. 2016.

8. Abu-Salma, Ruba, M. Angela Sasse, Joseph Bonneau, Anastasia Danilova, Alena Naiakshina, and Matthew Smith. "Obstacles to the adoption of secure communication tools." In *Security and Privacy (SP), 2017 IEEE Symposium on*, pp. 137-153. IEEE, 2017.